Your grade: 100%

Your latest: **100%** • Your highest: **100%** • To pass you need at least 80%. We keep your highest score.

Next item →

1. What is the effect of increasing the `"NumOctaves"` option when detecting SURF features? 1 / 1 point

- ○ More corners are detected.
- ○ Smaller blobs are detected.
- [✓ Larger blobs are detected.]

Correct
Yes! The `"NumOctaves"` option affects the sizes of the detected blobs. The larger the value, the larger the blobs.

2. Which of the following summarizes how the SURF extraction algorithm works for a single detected feature? 1 / 1 point

- [✓ It identifies the dominant orientation around a feature, identifies a square neighborhood around the feature oriented in this direction, and calculates gradient values in the neighborhood.]
- ○ It calculates gradient values in the neighborhood of the feature and computes a weighted average of these values using the orientation.
- ○ It identifies a circular neighborhood around the feature and averages gradient values of the pixels in the neighborhood.

Correct
Yes! These are the general steps of the SURF extraction algorithm. Other extraction algorithms follow a similar process.

3. How does the `extractFeatures` function in MATLAB choose an extraction algorithm by default? 1 / 1 point

- ○ It chooses FREAK for any corner feature input and SURF for any blob or region feature input.
- [✓ Based on the input object, it will choose either the FREAK, SURF, SIFT, ORB, or Block extraction algorithm.]
- ○ You need to specify which extraction algorithm you want to use.

Correct
Yes! You can see exactly which inputs result in which extraction algorithms in the "Feature Detection and Extraction Reference".

4. Some extraction algorithms are scale and rotation invariant, while others are not. Why might you use an algorithm that is not scale or rotation invariant? 1 / 1 point

- ○ You prioritize computational efficiency.
- ○ The images you work with are all taken from the same orientation.
- [✓ Both (a) and (b).]

Correct
Using a scale or rotation invariant algorithm results in more expensive computations. If you prioritize efficiency or don't need to account for changes in scale or rotation, you may choose to use an extraction algorithm like ORB or Block.

5. How is it determined whether two features match? 1 / 1 point

- [✓ By comparing the two feature descriptor vectors]
- ○ By checking if the pixel values in the neighborhoods surrounding the features are exactly the same
- ○ By comparing the x and y pixel locations of the two features

Correct
Yes! If the vectors are very similar, then the two features match.

Instructions for Questions 6-10

In questions 6-10, you will be working with the two stop sign images, found in your course files as "stop1.jpg" and "stop2.jpg". Open MATLAB and load them in using the following code:

```
img1 = imread("stop1.jpg");
img2 = imread("stop2.jpg");
```

6. Convert the images to grayscale and detect SURF features in both images using the default settings. How many SURF features were detected in the second image, `img2`? 1 / 1 point

- [✓ 757]
- ○ 50
- ○ 771

Correct
Yes, you can check the number of detected features using the `Count` property.

7. Once you have the detected features, you need to extract features from both images. What are the inputs and outputs of the **`extractFeatures`** function? 1 / 1 point

- [✓ Inputs: image and SURFPoints object (detected features)

  Outputs: extracted feature descriptions and SURFPoints object (extracted features)]
- Input: image

  Output: extracted feature descriptions
- Input: SURFPoints object (detected features)

  Output: SURFPoints object (extracted features)

Correct
Yes! Be sure to save both outputs of the function because you'll need them to perform matching and visualization.

8. Using the extracted features, you can now complete the feature matching and save the index pairs with the **`matchFeatures`** function. What is the 15th pair of indices (or the 15th row of the index pairs output)? 1 / 1 point

- [1 1]
- [14 18]
- [✓ [71 284]]

Correct
Yes! You can get the values of the 15th pair using

```
indexPairs(15,:)
```

9. Which of the following code blocks would you use to visualize your results, assuming you use the same variable naming conventions as those shown in the "Matching Features" video? 1 / 1 point

-
  ```
  showMatchedFeatures(img1,img2);
  ```
- [✓
  ```
  matchedPoints1=validPoints1(indexPairs(:,1),:);
  matchedPoints2=validPoints2(indexPairs(:,2),:);

  showMatchedFeatures(img1,img2,matchedPoints1,matchedPoints2);
  ```
  ]

-
  ```
  showMatchedFeatures(indexPairs);
  ```

Correct
Yes, you need to get the locations of the matched features, and then use them to visualize the result.

10. You've now completed the feature matching workflow! What do you think of the result? 1 / 1 point

The feature matching worked well, but some mismatches occurred. Refining feature extraction or using RANSAC for outlier removal could improve accuracy.

Feedback
Many of the features seem to be matched correctly, but there are definitely a few that are incorrect. In the next section of the course, you will learn a technique that can be used to identify incorrect matches.